Studies of an Event Building algorithm of the readout system for the twin TPCs in HFRS[*]

Jing Tian ,[1,2,3,4] Zhi-Peng Sun,[1,3] Song-Bo Chang,[1,3,4] Yi Qian,[1,3,4,†] Hong-Yun Zhao,[1,3,4] Zheng-Guo Hu,[1,3,4] and Xi-Meng Chen[2]

[1]*Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou 730000, China*
[2]*School of Nuclear Science and Technology, University of Lanzhou, Lanzhou 730000, China*
[3]*School of Nuclear Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China*
[4]*Advanced Energy Science and Technology, Guangdong Laboratory, Huizhou 516000, China*

The High Energy Fragment Separator (HFRS), which is currently under construction, is a leading international radioactive beam device. Multiple sets of position-sensitive Twin Time Projection Chamber (TPC) detectors are distributed on HFRS for particle identification and beam monitoring. The twin TPCs' readout electronics system operates in a trigger-less mode due to its high counting rate, leading to a challenge of handling large amounts of data. To address this problem, we introduced an event-building algorithm. This algorithm employs a hierarchical processing strategy to compress data during transmission and aggregation. In addition, it reconstructs twin TPCs' events online and stores only the reconstructed particle information, which significantly reduces the burden on data transmission and storage resources. Simulation studies demonstrated that the algorithm accurately matches twin TPCs' events and reduces more than 98% of the data volume at a counting rate of 500 kHz/channel.

Keywords: High counting rate, Twin TPCs, Trigger-less, Readout electronics, Event building, Hierarchical data processing

I. INTRODUCTION

With the ongoing development of superconducting heavy-ion accelerator technology, developed countries are proposing the construction of next-generation (third-generation) large scientific devices to obtain higher-intensity and higher-energy radioactive nuclear beams. They aimed to expand their research into heavier and closer-to-the-drip-line nuclide areas [1–3]. Currently, facilities such as BigRIPS [4] at RIKEN in Japan and ARIS [5] at MSU in the United States have been completed and are in operation, whereas others such as SuperFRS [6] at GSI in Germany and the High Energy Fragment Separator (HFRS) [7–10] at the Institute of Modern Physics, Chinese Academy of Sciences, are under construction. Once completed, the HFRS will be a new generation of radioactive beam devices with higher beam energy and magnetic rigidity ($B\rho$) and will significantly enhance the experimental capability in the medium-heavy nuclear region. Given the high-energy and high-intensity properties of HFRS, it is necessary to implement a highly reliable identification of nuclides at high counting rates (10 MHz) and meet the large dynamic range requirements ($Z = 1 \sim 92$), which poses a great challenge to radioactive particle detection and readout electronics technology. The Time Projection Chamber (TPC) [11–14] is a highly efficient and high-resolution detector for tracking particles that allows for the reconstruction of particle motion, momentum, and energy loss. It plays a crucial role in particle physics research and has been widely used in nuclear physics experiments such as the ALICE [15–17] Large Collision Experiment at CERN, the RHIC-STAR [18, 19] experiment at Brookhaven National Laboratory, and the CSR External-Target Experiment (CEE) [20, 21] currently under construction in China. Owing to its high reliability, the TPC has been chosen as a position-sensitive detector for particle identification and beam monitoring. However, as the counting rate of the projectiles increases, there is a strong possibility that the readout signals from different particles experience temporal confusion. For instance, when the counting rate is 10 MHz, the drift distance is 5 cm, and the drift velocity is 5 cm/us, particles are incident on the detectors per 100 ns, and the maximum drift time is 1 us. Under such circumstances, the probability of post-incident particle sensing signals on the readout strips before the pre-incident particle is as high as 90%. To address this issue, two TPC detectors with identical configurations placed in close proximity are inverted to form a set of twin TPCs [22], as illustrated in Fig. 1. When the same beam particle passes through the twin TPCs, the electrons generated by ionization eventually produce induced signals at the anode readout strips, which carry energy and time information. Once the readout electronics acquire and process the induction signal, the hit' position of the incident particle and the drift time of the electrons in the two TPCs are available for further analysis. In experiments with low counting rates such as GET (∼1 kHz) [23] and medium counting rates such as CEE (∼10 kHz) [24, 25], readout electronics employ the trigger-based method. The

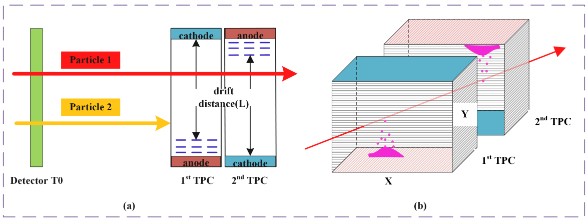


Fig. 1. (Color online)Illustration of the twin TPCs. (a) Schematic diagram of two beam particles incident on the twin TPCs; (b) An arrangement of TPC detectors placed in an inverted position.

[*] This work was partially supported by the Strategic Priority Research Program of Chinese Academy of Science (No. XDB 34030101) and the National Natural Science Foundation of China (Nos. 11975293 and 12205348).

[†] Corresponding author, qianyi@impcas.ac.cn

server cluster completes data processing tasks, such as transmission, event building, and storage. However, as the counting rate increases, the trigger-based method may no longer meet experimental demands. Therefore, a trigger-less readout scheme [26, 27] is proposed. For example, ATLAS (∼40 MHz total collision event rate) [28] and ALICE (∼50 kHz) [16] upgrade projects have adopted the trigger-less readout system, allowing the analysis and filtering of massive data in the back-end. Compared with the conventional trigger-based mode, the trigger-less system collects all the raw data from the front-end electronics, which improves the efficiency of obtaining valid events during the experiment, and the trigger algorithms are realized by the software, thus providing greater flexibility. However, trigger-less systems require significantly more bandwidth for transmission and processing, as well as increased computing power. These requirements increase development difficulty and overhead for readout systems. To overcome the bottleneck of data transmission and storage in trigger-less mode, and accurately match events from the twin TPCs, we propose an online algorithm for event building. Our algorithm employs a combination of software and hardware to handle massive amounts of data using a hierarchical processing strategy, and addresses the issue of particle time disorder at high counting rates. Specifically, data are compressed in each electronics readout unit and only key information is extracted, which significantly reduces the burden of transmission and storage. In addition, by utilizing information on the particle hit position and drift time obtained from the twin TPCs, we can effectively screen and reconstruct events, thereby recovering the original position information of the incoming particles. Consequently, our readout system exhibits a high counting rate and low data volume.

II. THE HIERARCHICAL DATA PROCESSING STRATEGY

The readout requirements for twin TPC are listed in Table 1. Even if the front-end electronics performs zero compression [29, 30], a high counting rate places enormous pressure on the data acquisition system. Therefore, it is imperative to explore a hierarchical processing strategy in each readout unit to reduce transmission and storage pressures. The readout system for the twin TPCs employs a three-tiered structural design consisting of front-end electronics (FEEs), slave data acquisition units (slave DAQs), and a server, which facilitate data processing at the channel, detector, and subsystem levels, respectively. The schematic in Fig. 2 illustrates the hierarchy of the data processing system. The master data acquisition unit (master DAQ) of HFRS collects data from each subsystem server for downstream processing and storage. Because our work focuses on the subsystem, we do not discuss the master DAQ in detail in this paper.

The main function of the FEE is to process the signals from the detectors at the channel level. The FEE components include front-end amplifiers for MWDC (FEAMs) [31, 32], which have been used in CEE, multichannel analog-to-digital converters (ADCs) (ADS52J90) [33] and a field-programmable gate array (FPGA) (Xilinx Kintex-7-325T) [34]. A FEAM chip can handle eight channels over a wide dynamic range (∼100 times). Each channel comprises a charge-sensitive amplifier (CSA), a shaper circuit, and a non-inverting driver circuit. The shaper circuit performs pole–zero cancellation, filtering, and shaping. The FEAM chip first amplifies and shapes the weak signals from the readout strips, and then passes the output signal to the ADC. The chip counting rate is currently being optimized for HFRS-TPC requirements. The design employs sub-board and mainboard configurations to facilitate readout capabilities for 32 or 64 channels. The sub-board comprises four FEAM chips, whereas the main board can house two sub-boards simultaneously. The FPGA firmware features an online algorithm for extracting time and energy information to minimize data volume.

TABLE 1. Twin TPC’s readout requirements.

Item	Value
Maximum total counting rate of the detector	10 MHz
Particle multiplicity	5
Number of readout channels	500
Single-channel average counting rate	∼100 kHz
Single-channel peak counting rate	∼500 kHz

The FPGA firmware of the FEE contains a self-triggering, data packaging, baseline subtraction, zero-compression, and a feature extraction modules. These modules cooperate to implement the initial data compression. The self-triggering module creates trigger signals for the data packing module at predetermined intervals, allowing the packing module to segment an uninterrupted input data stream based on these signals. Upon arrival of each trigger signal, a time window was initiated with a duration equal to the time interval between the two triggers. The relevant data within the time window were packaged in a particular format. To minimize the overhead caused by packet headers, a time window of 1000 ADC sampling intervals was utilized in this study. The counter of the system clock cycles is included in the packet header as a coarse timestamp supplied uniformly by the clock unit and calibrated at regular intervals. Subsequently, raw data are transmitted in packets. The baseline subtraction module adjusts the baseline by dynamically calculating the baseline noise using the currently available data. The zero-compression module discards invalid data by judging the validity of the pulse waveform and then sends valid pulses to the feature extraction module to obtain high-precision pulse time and energy information. Finally, the information for each pulse is repackaged and sent to the slave DAQ.

The slave DAQ executes the aggregation of detector-level data using a high-performance FPGA (Xilinx Kintex-UltraSCALE 060) [35], which aggregates and packages data from multiple FEEs with 10Gbps high-speed optical fiber links. In the trigger-less mode, the slave DAQ needs to sort the data in chronological order and channel order, which is convenient for downstream event screening and building. This was accomplished by developing a polling and sorting modules within the FPGA firmware of the slave DAQ. The polling module systematically extracts data from eight or 16

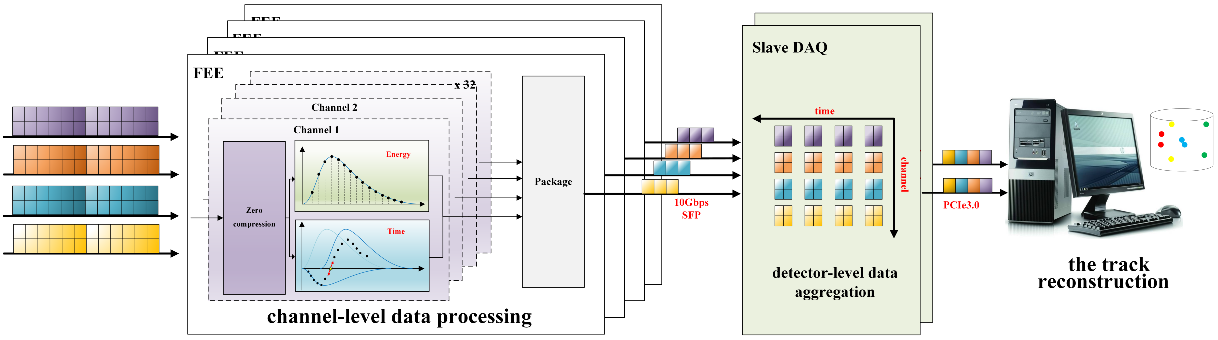


Fig. 2. (Color online)Schematic diagram of the data processing system hierarchy.

FEEs, unpacks the data packets, and recovers the original pulse information. The pulse information is then transmitted to the sorting module, which arranges and orders all the channel pulses using the time information. Finally, the slave DAQ packages the sequenced data and transmits them to the server through a PCIe interface.

The server plays a crucial role in issuing slow control instructions for the electronics system and reconstructing subsystem events, and the foremost step of event building involves using a time window to merge multiplicity signals. Once this was completed, the center of gravity (CG) approach was used to determine the original incident positions of the particles on the readout strips in each detector. Finally, both the incident position and drift time from the two TPCs were fed into the hit-matching module to reconstruct the particle tracks. The amount of data saved was significantly minimized by removing invalid data.

III. DESIGN AND IMPLEMENTATION OF EVENT BUILDING ALGORITHM

There are two types of interpretation of event building [36]. In brief, event building involves executing various software procedures on data, while, in broader sense, event building is determined by the system architecture of the software and hardware collaboration. Specifically, multilevel processing, including data packaging, feature extraction, and aggregation, is first performed on the FPGAs of each readout electronics unit. Then, software is leveraged to reconstruct physical events. Based on the hierarchical processing strategy outlined in Section II, the entire flow of the event-building algorithm includes signal preprocessing (adaptive baseline subtraction and zero compression), feature extraction (timing and energy extraction), multiplicity screening (fine timestamp sorting, fine timestamp merging, and CG for hitting position), and hit matching. Figure 3 illustrates the detailed algorithm pipeline and Fig. 4 displays the data format at each stage. A detailed description of the implementation of each module is provided below.

A. Signal pre-processing

To accommodate the discrepancies across channels, the channel-wise baseline and noise must be calculated within each packet. The baseline was determined by averaging the first 10 sample points of each channel, and the noise was calculated using the root-mean-square value of these points. In the selection process, if any sample point exceeds the minimum signal level required for electronics processing, the operation is stopped, and the region is deemed an effective signal segment. The baseline and noise were recalculated by moving back 20 points (adjusted for the pulse width). Considering both the baseline and noise, the threshold was measured as the sum of the baseline and four times the noise. If the sample point exceeded the threshold, the threshold was subtracted, whereas a sample point that was less than or equal to the threshold was replaced with zero.

Only signals with more than three consecutive sample points above the threshold were considered as valid pulse signals. The packet payload records the total quantity of these overthreshold sample points (Clust_L), the serial number of the first overthreshold points in the packet (Clust_T), and the corresponding data (D). Multiplying Clust_T by the ADC sampling interval (50 ns) gives the offset time of the pulse relative to the coarse timestamp in the packet header. Following zero compression, the total quantity of data (ChLen) in each channel should include all pulse data points and the corresponding descriptive information (Clust_L and Clust_T). This approach significantly reduced the amount of data.

B. Feature extraction

When operating in trigger-based mode, event screening is based on trigger signals; while in trigger-less mode, it relies on high-precision time information. Therefore, implementing a timing module is essential for enhancing the accuracy of the inferred Clust_T. The constant fraction discriminator (CFD) [37] divides the original signal into two paths: one delays the signal for a certain time, whereas the other inverts and weakens the signal. By merging these two paths, constant fraction timing is converted into zero-crossing timing, thereby effectively mitigating the effects of time wandering. Because

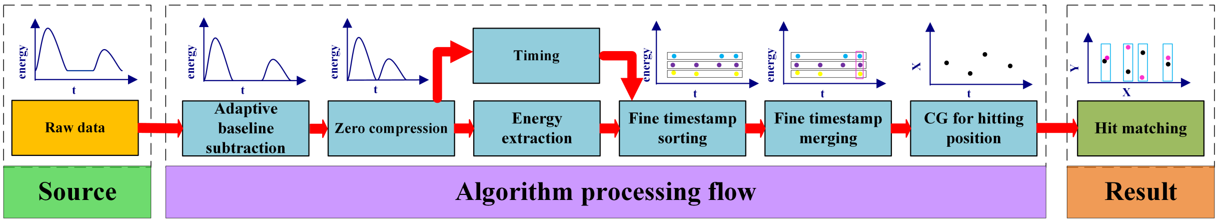


Fig. 3. (Color online)Workflow of the event building algorithm.

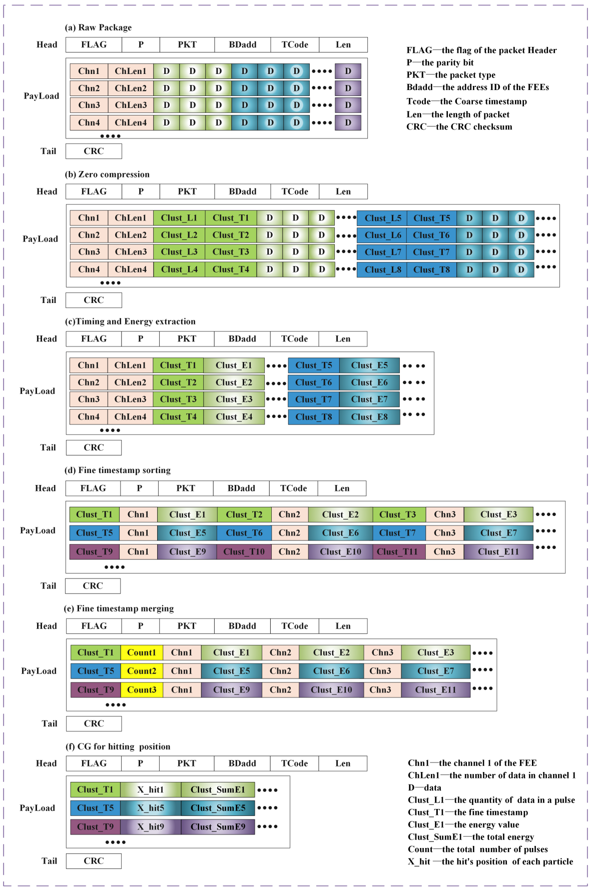

(a) Raw Package
(b) Zero compression
(c)Timing and Energy extraction
(d) Fine timestamp sorting
(e) Fine timestamp merging
(f) CG for hitting position
Head
PayLoad
Tail
FLAG
P
PKT
BDadd
TCode
Len
CRC
Chn1
ChLen1
D
Clust_L1
Clust_T1
Clust_E1
Count1
X_hit1
Clust_SumE1
FLAG—the flag of the packet Header
P—the parity bit
PKT—the packet type
Bdadd—the address ID of the FEEs
Tcode—the Coarse timestamp
Len—the length of packet
CRC—the CRC checksum
Chn1—the channel 1 of the FEE
ChLen1—the number of data in channel 1
D—data
Clust_L1—the quantity of data in a pulse
Clust_T1—the fine timestamp
Clust_E1—the energy value
Clust_SumE1—the total energy
Count—the total number of pulses
X_hit —the hit's position of each particle

Fig. 4. (Color online)Data packet format at each stage.

the ADC operates based on discrete sampling, the overthreshold time derived from CFD is often imprecise. Therefore, two data points were located before and after the overthreshold time (V_a and V_b), and a linear equation or interpolation technique was applied to achieve a more precise time measurement. During the initial validation of the software simulation, we derived a linear equation for the amplitude and time of the superimposed signals based on V_a and V_b: The exact moment of overthresholding corresponds to when the amplitude is zero. In the firmware, we use an interpolation method to approximate the threshold, which follows the principle of dichotomy [38]. We first calculated the mid-amplitude point (V_{mid}) based on V_a and V_b by dividing the original time interval into two equal sub-regions (a and b). The original time interval is equivalent to the ADC sampling interval. If V_{mid} multiplied by V_b is negative, we can determine that the overthreshold moment is in sub-region b; otherwise, it is in the other sub-region. Therefore, we reduced the time interval corresponding to the overthreshold moment to half that of the original. By repeating this process n times, we can increase the timing accuracy by a factor of 2^n compared to the original. Finally, the timing outcome was superimposed on Clust_T to provide a more precise pulse time. Clust_T is a 16-bit variable, and its high 10 bits are used to record the serial number of the data points within the packet (ranging from 0 to 1000) as the offset time relative to the coarse timestamp. The remaining six bits are utilized to record high-precision timing outcomes, which subdivides 50ns into 64 equal parts, thereby improving the time resolution.

Events can accumulate at high counting rates. The peak method can better distinguish overlapping signals compared to the area integration method. When several adjacent sample points first increase and then decrease, the peak (Clust_E) is at the vertex. After the energy extraction, only Clust_T and Clust_E were recorded for each pulse. We used the ChLen field to denote the pulse count for each channel to facilitate subsequent sorting.

C. Multiplicity screen

When particles enter the detector, there is usually an angle between the incoming direction and detector plane. Transverse diffusion occurs after the particle incidence, enabling multiple readout strips to produce induction signals. The signals have consistent timestamps, but different levels of magnitude across multiple readout channels, which is called particle multiplicity [22] as shown in Fig. 5(a). Thus, a multiplicity screen was required to recover the original hit position of the particle. When merging the particle multiplicity, the first step is to arrange the pulses of all channels within the packet in chronological and channel order. Because all the pulses in a packet share the same coarse timestamp (TCode), they only need to be sorted according to Clust_T, which is known as fine timestamp sorting. Figure 5(b) shows the workflow of fine timestamp sorting. Since the pulse data of each channel are stored in separate containers (FIFOs or arrays) in chronological order, only the first pulse data in each container were considered for comparison. Subsequently, the pulse data corresponding to the smallest moment obtained from this comparison were moved from the old container to the new con-

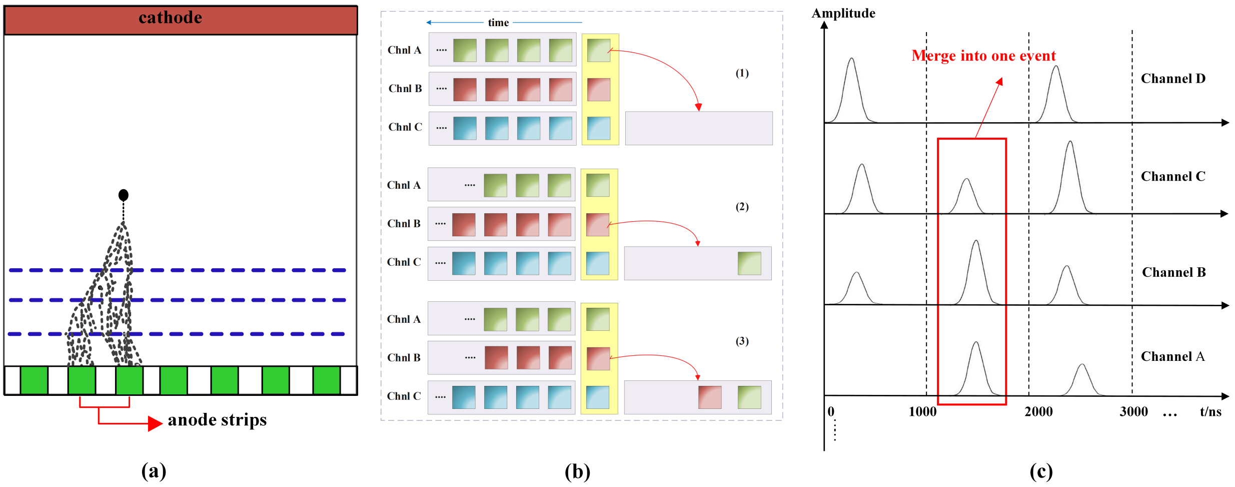


Fig. 5. (Color online)Schematic diagram of (a) Particle multiplicity,(b) Fine timestamp sorting, and (c) Fine timestamp merging.

tainer. In the second round, the second pulse data from the old container were compared with the first pulse data from the other containers. This process continues until all data in each container are compared and stored in the new container. If the pulse times in the two containers are equal, they are arranged in the channel order. If the time interval between the adjacent channel's pulse signals is shorter than the designated time window, they are categorized as the same event and can be merged according to the channel order, as illustrated in Fig. 5(c). The Clust_T value was determined based on the fine timestamp of the central channel, and a count field was added to record the total number of pulses in each event. This information is then used in the CG to identify the hit's position of the incident particle.

Using CG (in Eq. (1)), the hit's position (X_{hit}) of each particle on the readout strip was determined based on the previously obtained channel-fired ID and energy value. The channel-fired ID was used as the x_i coordinate, and the energy value acted as the weight E_i. The drift time of the electrons is measured as the sum of the coarse timestamp and fine timestamp. Both the hit's position and drift time from the twin TPCs were relayed to the hit-matching module, and the two-dimensional trace of the incoming particles was ultimately reconstructed. This process also helps discard invalid or irrelevant data.

$$X_{\text{hit}} = \sum x_i \cdot E_i / \sum E_i \tag{1}$$

D. Hit matching

In Eq. (2), the drift distance (L) and drift speed (v_{drift}) of electrons for twin TPCs are predetermined. As a result, the sum of the drift time (t_{cs}) remains fixed and can be used as a constraint to achieve hit-matching [22] of twin TPCs. Although the plastic scintillator detector (T_0) can provide a highly accurate reference time (t_0), a large fluctuation in the particle flight time can negatively impact its accuracy. Moreover, t_0 information is transferred directly to the master DAQ, and system-wise event building is achieved by combining the data from the subsystem servers, which may reduce timeliness. Considering the aforementioned factors, we propose a novel hit-matching algorithm for reconstructing particle tracks without t_0 information. This allows event building to be performed on the subsystem server, thus enabling additional data compression and easing the computational burden on the master DAQ.

$$t_{\text{cs}} = t_u + t_d - 2t_0 = \frac{L}{v_{\text{drift}}} \tag{2}$$

where t_u, t_d is the absolute drift time of electrons from the twin TPCs and t_0 is the reference time from detector T_0.

The matching algorithm can be classified into two methods based on whether the factor t_0 is incorporated: the relative time method and the absolute time method.

1) the relative time method

(a) Search for all hits from twin TPCs in the range $[t_0 - 5\sigma_t, t_0 + t_{\text{dmax}} + 5\sigma_t]$ according to t_0.

(b) Time matching: For the searched hits, use their drift times and t_0 to calculate t_{cs}, then select the hit combinations within $5\sigma_{\text{tcs}}$.

(c) Position matching: For hits that meet the time matching conditions, choose the combination with the smallest hit distance.

(d) Positioning: According to the drift time of the chosen hit, combined with the drift distance and drift speed, calculate the hit position in the Y direction.

2) the absolute time method

(a) Search for all all hits from the second TPC in the range $[t_u - t_{\text{dmax}} - 6\sigma_t, t_u + t_{\text{dmax}} + 6\sigma_t]$ according to t_u.

(b) Time matching: For the searched hits, calculate $|t_u - t_d|$ and select the hit combinations within $12\sigma_t$.

(c) Position matching: For hits that meet the time matching conditions, choose the combination with the smallest hit distance and minimize the energy difference.

(d) Positioning: According to the drift time of the chosen hit, combined with the drift distance and drift speed, calculate the hit position in the Y direction.

$$\sigma_{t_{cs}} = \sqrt{2{\sigma_t}^2 + 4{\sigma_{t_0}}^2} \tag{3}$$

where $\sigma_t, \sigma_{t_0}, \sigma_{\text{tcs}}$ are the time resolutions of the TPC detector, T_0 detector, and t_{cs} variable, respectively.

IV. VERIFICATION AND ANALYSIS

As the development of TPC and front-end electronics is underway, we were unable to test the algorithm on the corresponding devices. Alternatively, we first conducted a simulation with software to verify the performance of the proposed event-building algorithm and then performed preliminary testing with existing laboratory equipment.

A. Simulation study

We use the Monte Carlo method [39] to generate the twin TPCs' simulation packet. When particles enter the twin TPCs, electrons generated by ionization drift up and down, with the drift time based on their hits' position in the Y direction, the total drift distance remains unchanged. The fired readout channels were determined by the hits' position on the readout strips, incidence angle, and transverse diffusion, with a maximum of five channels. Therefore, the drift time and fired channels can be set according to the length and width of the TPCs' incidence cross section. The FEAM chip exhibited a peak time of 160 ns and a falling edge at approximately 320 ns. As a result, the $CR-(RC)^3$ characteristic equation is employed to derive accurate pulse data, and the signal amplitude range is defined by the selected ADC bit width. When the system's counting rate reaches 1 MHz, a particle is expected to hit the twin TPCs every 1000 ns on average. Therefore, two sets of simulation waveform data were simultaneously produced every 1000 ns (t_0). The amplitude, fired channels, and drift time (with t_0 as the reference) for each waveform were randomly generated to mimic a real scenario accurately. Baseline noise obtained from existing FEE measurements was added to the simulation data to enhance authenticity. In addition, simulation data were produced for each TPC using only one FEE consisting of 32 channels.

Figure 6 shows the overall distribution of the simulation data for a pair of complete packets for the twin TPCs. Multiple simulation waveforms were created within a 50us period, traversing 32 different channels with varying peak amplitudes and covering most of the ADC range (bit width of 10). Figure 7(a) shows the simulated waveforms of the twin TPCs' central channel fired at a specific time, which exhibit high similarity with only minor amplitude variations. The phase relationship between the two drift times covers the three cases of overrun, approach, and lag that satisfy the simulation requirements. Furthermore, Figure 7(b) shows the simulated waveforms after removing the baseline, which dropped to zero. The two TPC packets contained 219 and 210 pulses, respectively. In other words, an average of 6∼7 pulses were generated on each readout channel in 50us time for both TPCs, so the average counting rate greater than 100 kHz/channel as expected, indicating that the simulation data closely resembles the actual situation.

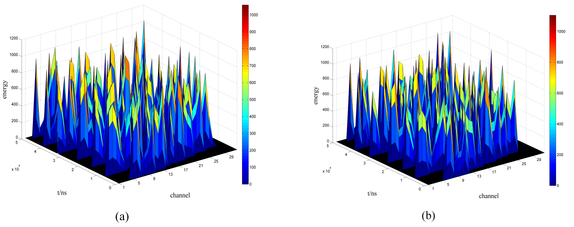


Fig. 6. (Color online)Distribution of (a) The 1st TPC' simulation data and (b) The 2nd TPC' simulation data.

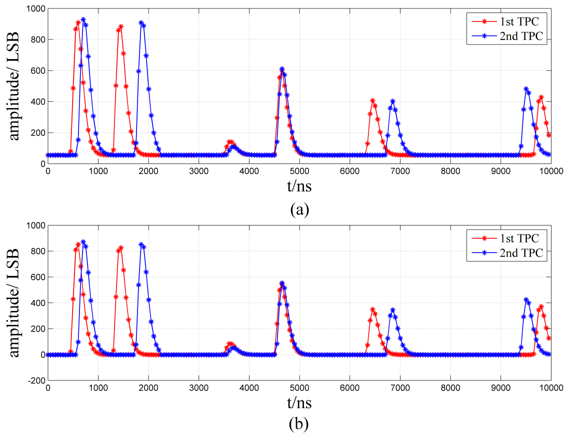


Fig. 7. (Color online) Simulated waveforms of the twin TPCs' central channel fired. (a) Raw simulated waveforms; (b) Waveforms after subtracting the baseline.

After merging the multiplicities, the first TPC collected 50 sets of event data and the second TPC recorded 49 sets of event data. Subsequently, the hits' positions and drift times of all event data are transmitted to the hit-matching module, where the absolute time method is employed for matching. 49 data sets were successfully matched, and only one event dataset was discarded due to matching failure. Finally, based on the matched drift time pairs, the drift distances were calculated, and the particle hit positions in the Y-direction were recovered, as depicted in Fig. 8. When the incident particle strikes the twin TPCs, three drift distances are possible: the first TPC's drift distance may be greater or less than that of the second TPC, or the two drift distances may be equal; however, the sum of the drift distances in the two TPCs is always the maximum drift distance. Figure 8 shows three typical cases: (a) when the incident particle collides with the center of the twin TPCs and the two drift distances are equal; (b) when the particle impacts a position close to the first TPC's readout strip, the second TPC experiences the maximum drift distance; and (c) if the hit is offset to the second TPC's readout strip, the first TPC undergoes a larger drift distance than the second TPC. Theoretically, the particle should hit the same position on the readout strips of both TPCs. However, in practice, a slight deviation between the two hits' positions is possible due to oblique incidence. Using the relative-time approach, the same datasets produced consistent results, as shown in Fig. 8.

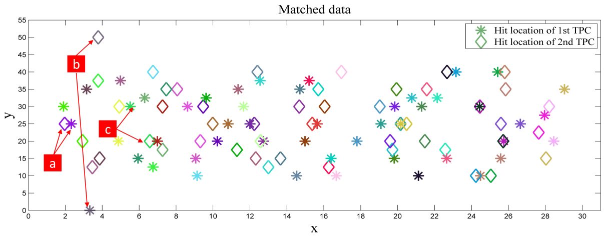
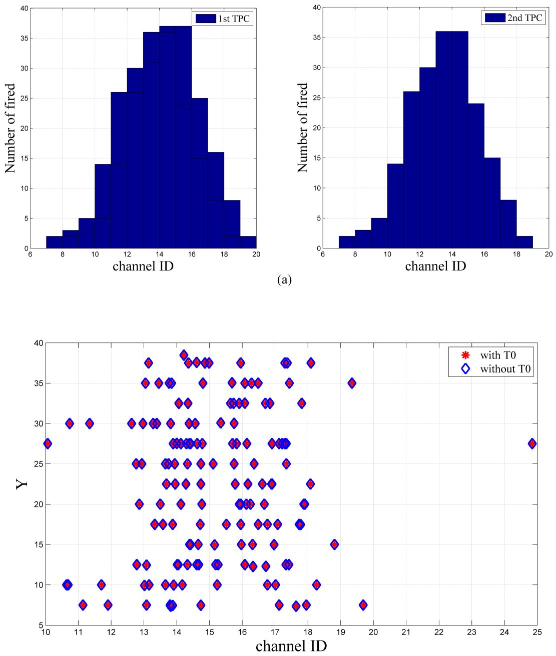


Fig. 8. (Color online)Matched results obtained using the absolute time method from the twin TPCs' simulation packets. (Two different symbols of the same color represent the hits' positions of the same incident particle in two TPCs, respectively.)

To verify the performance of event-building algorithm at a high counting rate, the hit probability of the readout channel was adjusted from a uniform to Gaussian distribution. Furthermore, the probability of the readout channels in the central region being fired increases. As illustrated in Fig. 9(a), four to five channels within the two TPC packets were fired more than 25 times in 50us, indicating a counting rate of 500 kHz/channel. Three sets of packets were analyzed and the matching results obtained using both methods were consistent, as shown in Fig. 9(b). It was observed that both matching algorithms could effectively recover the two-dimensional traces of incident particles under normal conditions. However, the relative time method is more suitable for severe cases because it employs a more rigorous equation as a constraint. Therefore, the absolute time method can be utilized in the subsystem server to initially match hits, remove inconsistent data, and further reduce data volume. In addition, the relative-time technique can be employed in the master DAQ to accomplish system-wise hit identification.

As an illustration, the current simulation parameters, that is, a single data packet containing 32 channels with 1000 sample points each and an additional 16-bit descriptive information as the raw data volume were used to evaluate the data compression capability for the event-building algorithm. The evaluation was based on the following assumptions.



(a)

(b)

Fig. 9. (Color online) (a) Number of fired for each channel in the twin TPCs; (b) Particles hit positions in the Y-direction obtained by hit matching in the 1st TPC: The red symbol represents the results acquired by the relative time method, while the blue symbol shows the outcomes obtained using the absolute time method.

1) A total of 50 particles are incident in 50 microseconds and each particle hits 5 readout channels, generating a total of 250 valid pulses.

2) After removing the baseline, an average of 15 sampling points were retained for each valid pulse.

3) The fine timestamps and energy information were each 16 bits.

4) The position information for the hits was 10 bits in both the X (readout strip) and Y directions.

The compression ratios for several stages are listed in Table 2, which indicates that the event-building algorithm can shrink 99% of the raw data and 97% of the zero-compressed data.

B. Test Verification

As shown in Fig. 10, a primary test system was set up in the laboratory using existing electronic devices. Note that the chassis of the remote server and slave DAQ contained within it are not visible in this figure. A signal generator (keysight33522B) [40] was employed to create a trigger signal and two-pulse signals. The two pulse signals were fed into the two FEAM chip's analog inputs on the FEE to emulate the signals from two TPCs, respectively. The input signals were first converted into digital form by the ADC, and then transmitted using an optical fiber to the slave DAQ for aggregation and packaging. Finally, the data were sent to a remote server

TABLE 2. The compression ratios for several major processing phases.

Data-processing phase	The data volume of the packet payload	Compression ratio
Raw data	320512 bits	100%
Zero compression	43512 bits	14%
Timing and energy extraction	8512 bits	3%
CG for hitting position	2100 bits	$<1\%$
Hit matching	1000 bits	$<1\%$

via a PCIe interface. The drift time of the electrons generated by the primary ionizing particles was simulated by adjusting the delay time of the input signal relative to the trigger signal. Multiple pulse signals collected by the FEE were then spliced together to mimic a trigger-less mechanism. With a sampling frequency of 50 MHz, the ADC collects 100 samples per pulse signal, implying a pulse period of 2 μs and a counting rate of 500 kHz/channel. Figure 11(a) illustrates the initial waveforms of the two FEE channels' output at specific times. Next, the adaptive baseline subtraction module analyzes and removes any baselines from the data. The outcomes of this procedure are shown in Fig. 11(b). The results indicate the effectiveness of the module in cleaning the baseline noise.

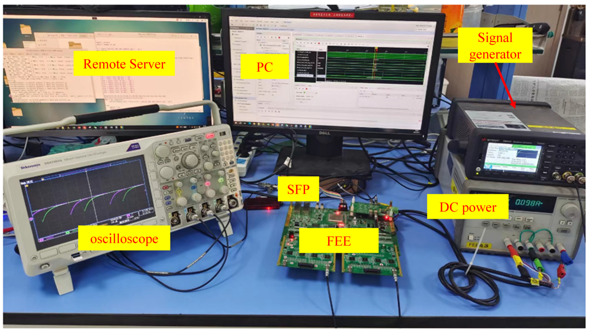


Fig. 10. (Color online) Electronics test system.

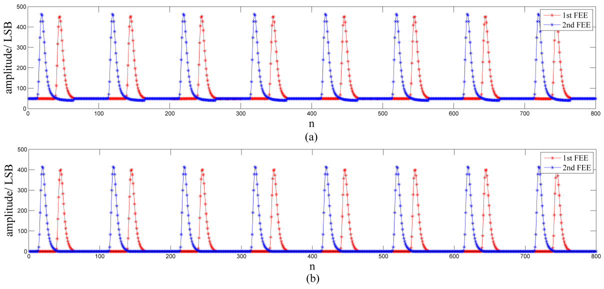


Fig. 11. (Color online) (a) Original waveforms; (b) Waveforms after subtracting the baseline.

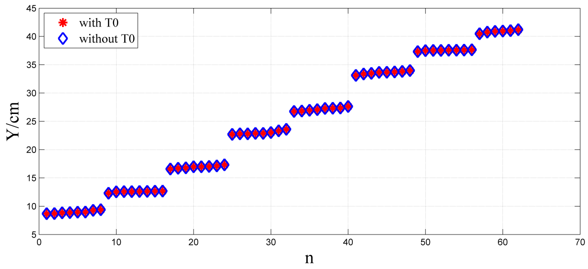


Fig. 12. (Color online)Particles hit positions in the Y-direction obtained by hit matching in the 1st TPC: The red symbol represents the results acquired by the relative time method, while the blue symbol shows the outcomes obtained using the absolute time method.

To verify the performance of the event-building algorithm using experimental data, we selected eight typical delay time pairs and five different amplitude combinations of input signals to conduct 40 tests, each consisting of eight pulses. The additional delay caused by the readout system must be subtracted before data processing. After processing, 320 test pulses were sent to the hit matching module, all of which were accurately matched using either absolute or relative time methods. The Y-direction positions where the particles hit were determined using 320 pairs of drift times. From this dataset, 20% of the data were extracted and plotted in Fig. 12, which shows that the results of the two matching algorithms were highly consistent. The key test parameters and compression ratios are shown in Table 3. The preliminary evidence shows that the event-building algorithm can effectively compress data.

TABLE 3. The key parameters and compression ratios

Item	Value
Counting rate	∼500 kHz/channel
Number of channels	1
Pulse width (ns)	∼500
ADC sampling rate (MHz)	50
The raw packet payload	8016 bits (100%)
The packet payload after zero compression	2192 bits (∼28%)
The packet payload after hit-matching	160 bits (∼2%)

V. CONCLUSION

This paper presents an event building algorithm that aims to tackle two major challenges: the transmission and storage of large amounts of data generated by the trigger-less readout system, as well as the issue of particle time disorder at high counting rates. The algorithm conducts hierarchical processing and compression of data on FEEs, slave DAQs, and a server platform, and finally matches the twin TPCs' events on the server to reconstruct the two-dimensional traces of incident particles. The processing flow of the algorithm was fully implemented by software. The data generated from both the simulation and preliminary laboratory tests were accurately matched using the algorithm, and the results suggested that it could effectively reduce the data volume by at least 98%. Potential future directions for improvement include implementing firmware logic for timing, energy extraction, and sorting by utilizing FPGAs on both FEEs and slave DAQs. This leverages the pipeline parallelism and low latency of the FPGA to enhance the overall system performance. In addition, the use of GPU multicore parallel computing [41, 42] to improve the speed of multiplicity merging and hit matching modules is also a direction for future research.

AUTHOR CONTRIBUTIONS

All authors contributed to the study conception and design. The algorithm implementation and material preparation were performed by Jing Tian and Zhi-Peng Sun. The data collection and analysis were performed by Jing Tian and Song-Bo Chang. The first draft of the manuscript was written by Jing Tian and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

CONFLICT OF INTEREST

The authors declare that they have no competing interests.

[1] Y. Zhang, D.W. Zhang, X.F. Luo, Experimental study of the QCD phase diagram in relativistic heavy-ion collisions. Nucl. Tech. **46**, 040001 (2023). doi:10.11889/j.0253-3219.2023.hjs.46.040001

[2] Q. Chen, G.L. Ma, J.H. Chen, Transport model study of conserved charge fluctuations and QCD phase transition in heavy-ion collisions. Nucl. Tech. **46**, 040013-040013 (2023). doi: 10.11889/j.0253-3219.2023.hjs.46.040013

[3] K. Wang, H.S. Song, F. Fang et al., Anti-coincidence detector unit based on WLS fiber and SiPM readout. Nucl. Tech. **46**, 070401 (2023). doi: 10.11889/j.0253-3219.2023.hjs.46.070401

[4] H. Takeda, Y.K. Tanaka, H.Suzuki et al., Exotic nuclei and their separation, ed. by Isao.Tanihata, Hiroshi.Toki (Springer, Singapore, 2020), pp.1–59. https://doi.org/10.1007/978-981-15-8818-1_100-1

[5] M. Hausmann, A.M. Aaron, A.M. Amthor et al., Design of the Advanced Rare Isotope Separator ARIS at FRIB. Nucl. Instrum. Methods Phys. Res.B **317**, 349–353 (2013). https://doi.org/10.1016/j.nimb.2013.06.042

[6] E. Haettner, B. Franczak, H. Geissel et al., New high-resolution and high-transmission modes of the FRS for FAIR phase-0 experiments. Nucl. Instrum. Methods Phys. Res. B **463**, 455–459 (2020). https://doi.org/10.1016/j.nimb.2019.04.025

[7] Y. Yang, Y.W. Su, W.Y. Li et al., Evaluation of radiation environment in the target area of fragment separator HFRS at HIAF. Nucl. Sci. Tech. **29**, 147 (2018). https://doi.org/10.1007/s41365-018-0479-9

[8] J.C. Yang, J.W. Xia, G.Q. Xiao et al., High Intensity Heavy Ion Accelerator Facility (HIAF) in China. Nucl. Instrum. Methods Phys. Res. B **317**, 263–265 (2013). https://doi.org/10.1016/j.nimb.2013.08.046

[9] L.N. Sheng, X.H. Zhang, J.Q. Zhang et al., Ion-optical design of High Energy Fragment Separator (HFRS) at HIAF. Nucl. Instrum. Methods Phys. Res. B **469**, 1–9 (2022). https://doi.org/10.1016/j.nimb.2020.02.026

[10] X.H. Zhou, J.C. Yang and the HIAF project team, Status of the high-intensity heavy-ion accelerator facility in China. AAPPS Bulletin **32**, 35 (2022). http://dx.doi.org/10.1007/s43673-022-00064-1

[11] D. Attie, TPC review. Nucl. Instrum. Methods Phys. Res. A **598**, 82–93 (2009). https://doi.org/10.1016/j.nima.2008.08.114

[12] A. Delbart, Development of resistive micromegas TPCS for T2K experiments. Nucl. Instrum. Methods Phys. Res. A **958**, 162-798 (2020). https://doi.org/10.1016/j.nima.2019.162798

[13] K. Ackermann, S. Arai, A study with a small prototype TPC for the international linear collider experiment. Nucl. Instrum. Methods Phys. Res. A **623**, 141-143 (2010). https://doi.org/10.1016/j.nima.2010.02.175

[14] H.K. Wu, Y.J. Wang, Y.M. Wang et al., Machine learning method for ^{12}C event classification and reconstruction in the active target time-projection chamber. Nucl. Instrum. Methods Phys. Res. A **1055**, 168528 (2023). https://doi.org/10.1016/j.nima.2023.168528

[15] The ALICE Collaboration, The ALICE Experiment at the CERN LHC. J. Instrum. **3**, S08002 (2008). https://dx.doi.org/10.1088/1748-0221/3/08/S08002

[16] W.H. Trzaska (ALICE Collaboration), New ALICE detector for Run 3 and 4 at the CERN LHC. Nucl. Instrem. Methods Phys. Res. A **958**, 162116 (2020). https://doi.org/10.1016/j.nima.2019.04.070

[17] A. Di. Manro, New inner tracking system for ALICE Upgrate at the LHC. Nucl. Instrum. Methods Phys. Res. A **936**, 625-629 (2019). https://doi.org/10.1016/j.nima.2018.10.047

[18] F. Shen, S. Wang. MWPC prototyping and performance tests for STAR inner TPC upgrades. Nucl. Instrum. Methods Phys. Res. A **896**, 90-95 (2018). https://doi.org/10.1016/j.nima.2018.04.019

[19] K.H. Ackermann, N. Adams, STAR detector overview. Nucl. Instrum. Methods Phys. Res. A **499**, 624–632 (2003). https://doi.org/10.1016/S0168-9002(02)01960-5

[20] L. Lü, H. Yi, Z. Xiao et al., Conceptual design of the HIRFL-CSR external-target experiment. Sci. China Phys. Mech. Astron. **60**, 012021 (2017). https://doi.org/10.1007/s11433-016-0342-x

[21] W. Huang, F. Lu, H. Li et al., Laser test of the prototype of CEE time projection chamber. Nucl. Sci. Tech. **29**, 41 (2018). https://doi.org/10.1007/s41365-018-0382-4

[22] F. García, T. Grahn, J. Hoffmann et al., A GEM-TPC in twin configuration for the Super-FRS tracking of heavy ions at FAIR. Nucl. Instrum. Methods Phys. Res. A **884**, 18–24 (2018). https://doi.org/10.1016/j.nima.2017.11.088.

[23] S. Anvar et al., AGET, the GET front-end ASIC, for the readout of the Time Projection Chambers used in nuclear physic experiments.IEEE Nuclear Science Symposium Conference

Record. Valencia. Spain. 745-749 (2011). doi: 10.1109/NSS-MIC.2011.6154095

[24] T.X. Wang, Research and Implementation of the DAQ System for CEE. Dissertation, University of Science and Technology of China, 2022. (in Chinese)

[25] H.L. Wang, Z. Wang, C.S. Gao et al., Design and tests of the prototype beam monitor of the CSR external target experiment. Nucl. Sci. Tech. **33**, 36 (2022). https://doi.org/10.1007/s41365-022-01021-1

[26] X. Gao, D. Emschermann, J. Lehnert et al., Throttling strategies and optimization of the trigger-less streaming DAQ system in the CBM experiment. Nucl. Instrum. Methods Phys. Res. A **978**, 164442 (2020). https://doi.org/10.1016/j.nima.2020.164442.

[27] M. Jubin, D. Erno, M. Eduardo et al., Trigger and timing distributions using the TTC-PON and GBT bridge connection in ALICE for the LHC Run 3 Upgrade. Nucl. Instrum. Methods Phys. Res. A **922**, 119-133 (2019). https://doi.org/10.1016/j.nima.2018.12.076

[28] Jörn Schumacher, Improving packet processing performance in the ATLAS FELIX project: analysis and optimization of a memory-bounded algorithm.DEBS '15: Proceedings of the 9th ACM International Conference on Distributed Event-Based Systems. pp 174—180 (2015). https://doi.org/10.1145/2675743.2771824.

[29] B.L. Hou, Research on the prototype electronics of RICH PID in Super Tau-Charm Facility. Dissertation, University of Science and Technology of China, 2022. (in Chinese)

[30] B.L. Hou, L. Zhao, J.J. Qin et al., Prototype of the readout electronics for the RICH PID detector in the STCF. Nucl. Sci. Tech. **33**, 80 (2022). https://doi.10.1007/s41365-022-01056-4

[31] Z.H. Jia, Y. Qian, T.L. Pu et al., The design of multi-channel low noise preamplifier. Nucl. Electron. Detect. Technol. **38**, 65–69 (2018).

[32] T.L.Pu, The design and research on key technology of front-end readout ASIC chips for gas detector. University of Chinese Academy of Sciences (Institute of Modern Physics, Chinese Academy of Sciences), 2020 (in Chinese).

[33] ADS52J90 Data Sheet.https://www.ti.com/lit/ds/symlink/ads-52j90.pdf. Accessed 18 Apr 2018

[34] Kintex-7 FPGAs Data Sheet.https://docs.xilinx.com/v/u/en-US/ds182_Kintex_7_Data_Sheet.Accessed 26 Mar 2021

[35] Kintex UltraScale FPGAs Data Sheet. https://docs.xilinx.com/r/en-US/ug571-ultrascale-selectio/UltraScale-Architecture-SelectIO-Resources-User-Guide. Accessed 22 Sept 2020

[36] L. Zhang, J. Yang, T. Wang et al., Event building algorithm in a distributed stream processing data acquisition platform: D-Matrix. IEEE Trans. Nucl. Sci. **70**, 105-112 (2023). https://doi.org/10.1109/TNS.2023.3235904

[37] W.W. Ding, C.H. Hu,Y. Cao et al., Implementation of digital constant fraction timing based on field programmable gate array. J. Phys. Conf. Series **1941**, 012068 (2021). https://doi.org/doi: 10.1088/1742-6596/1941/1/012068

[38] X.Q. Zhang, J.B. Fang, C.S. Lin et al., Complexity analysis of dichotomy method in solving the approximated solution of equations. IEEE International Symposium on IT in Medicine and Education, Cuangzhou, pp 257-259, 2011. https://doi: 10.1109/ITiME.2011.6130828

[39] G. Romeo, Elements of Numerical Mathematical Economics with Excel. Academic Press, 2020, pp 763–795. https://doi.org/10.1016/B978-0-12-817648-1.00014-1

[40] 33522B Waveform Generator, 30 MHz, 2-Channel with Arb. https://www.keysight.com/us/en/product/33522B/waveform-generator-30-mhz-2-channel-arb.html.Accessed 25 July2023.

[41] J. Fang, C. Huang, T. Tang et al., Parallel programming models for heterogeneous many-cores: a comprehensive survey. CCF Trans. HPC **2**, 382–400 (2020). https://doi.org/10.1007/s42514-020-00039-4

[42] G. Michael, K. Ben, A hybrid CPU/GPU approach for optimizing sorting throughput. Parallel Computing **85**, 45–55 (2019). https://doi.org/10.1016/j.parco.2019.01.004